

1/13

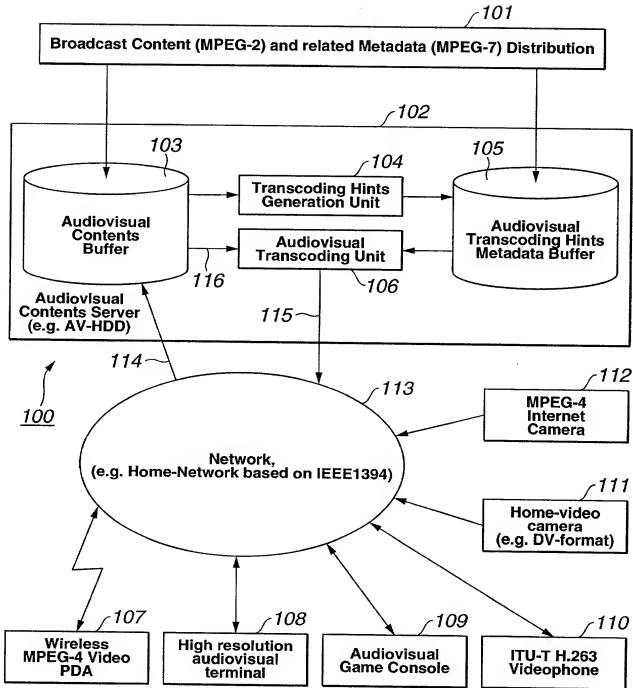


FIG.1

2/13

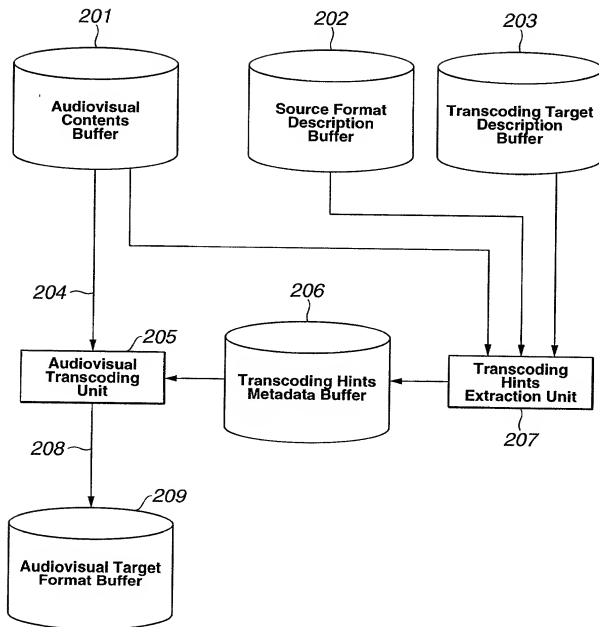


FIG.2

3/13

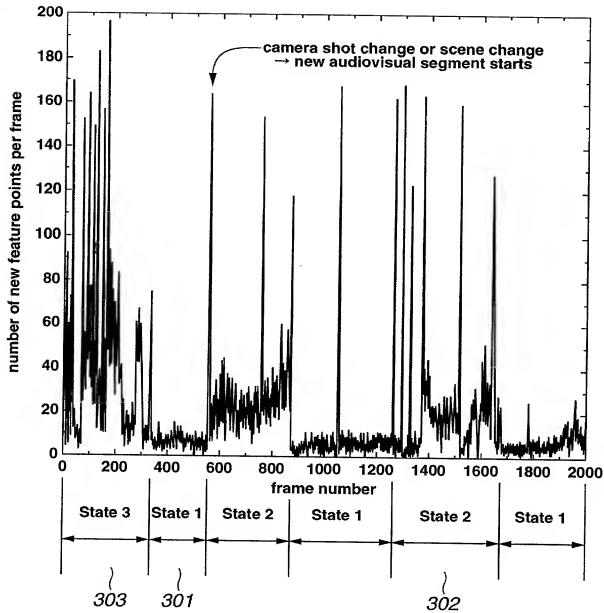


FIG.3

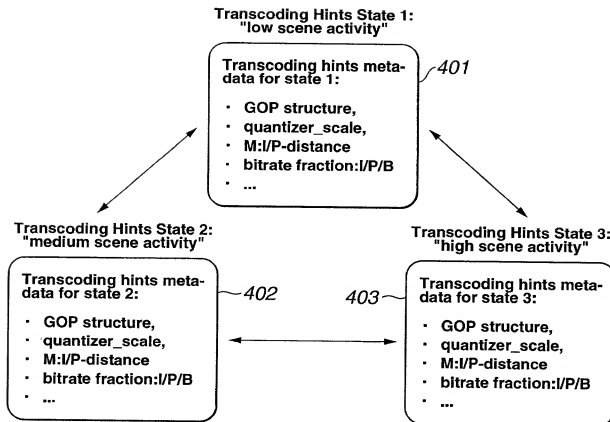


FIG.4

5/13

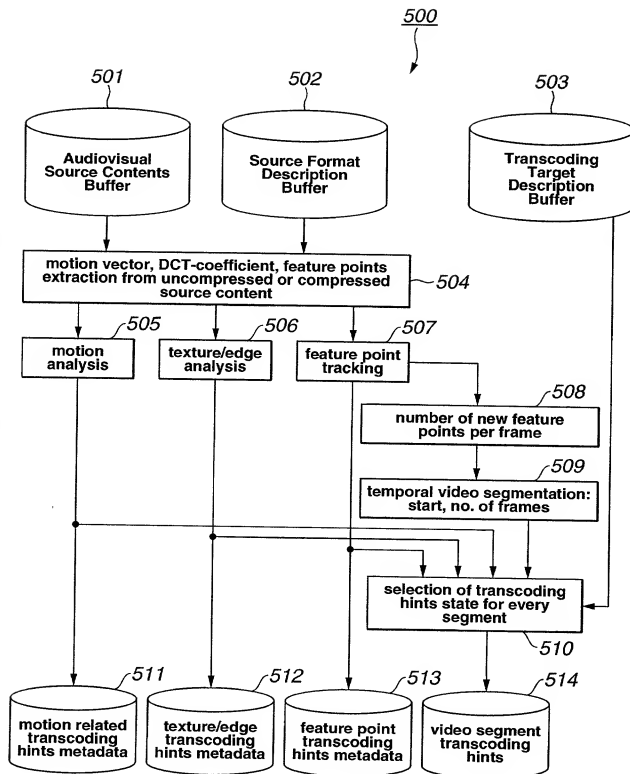


FIG.5

6/13

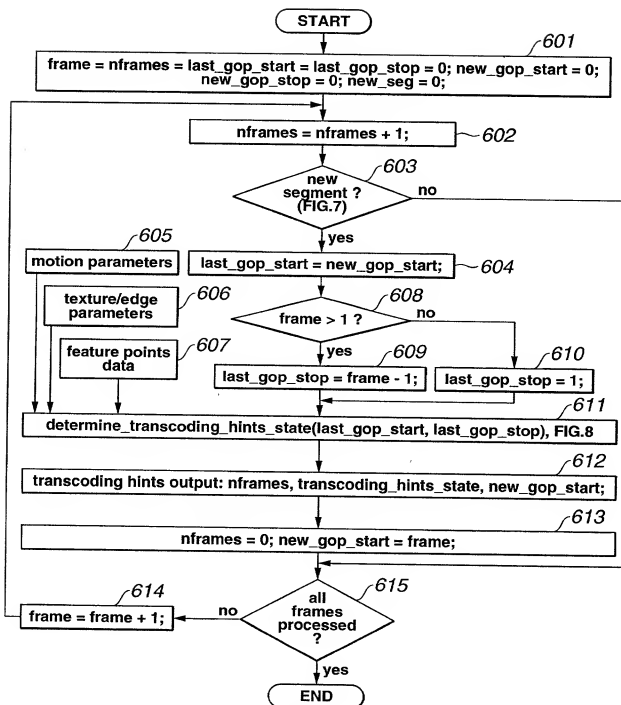


FIG. 6

7/13

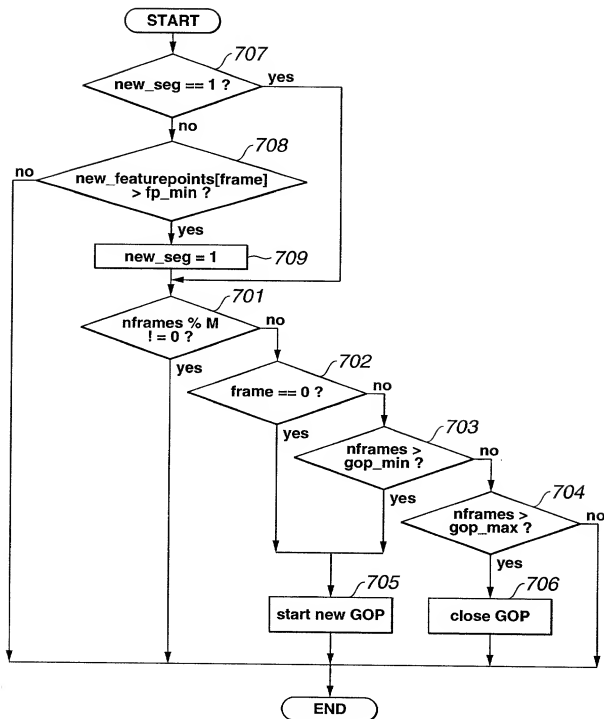


FIG.7

8/13

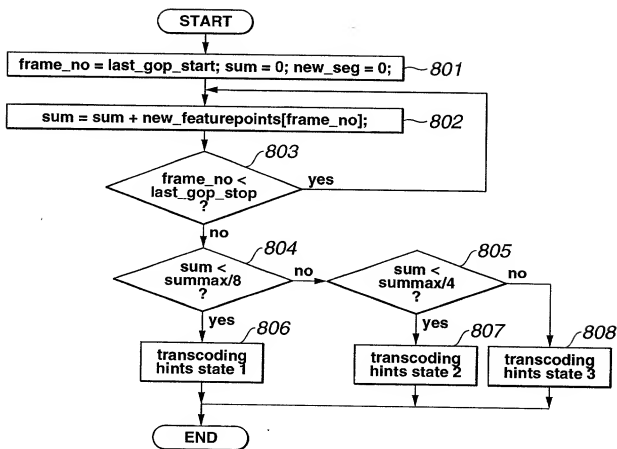


Table 1: Transcoding Hints State Table (example)

State	M	bitrate_fraction_for_l	bitrate_fraction_for_p
1	5	0.8	0.15
2	4	0.85	0.1
3	3	0.9	0.05

FIG.8

9/13

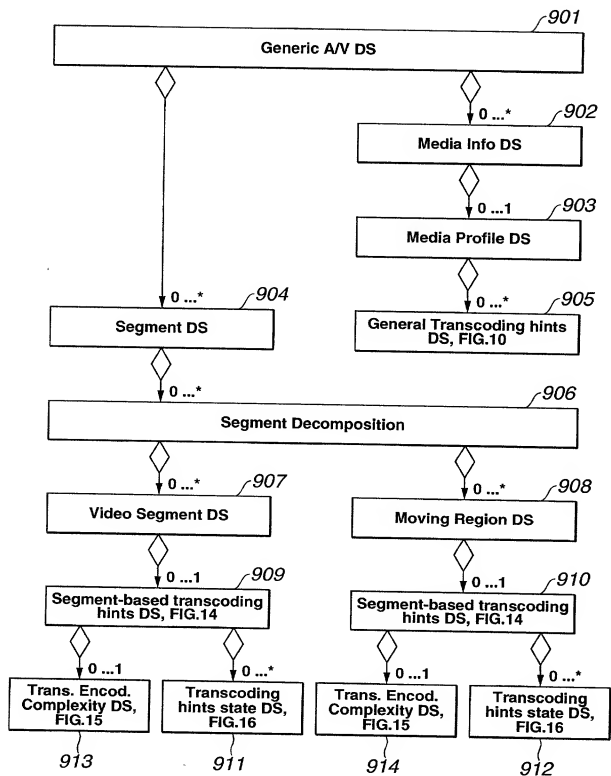


FIG. 9

10/13

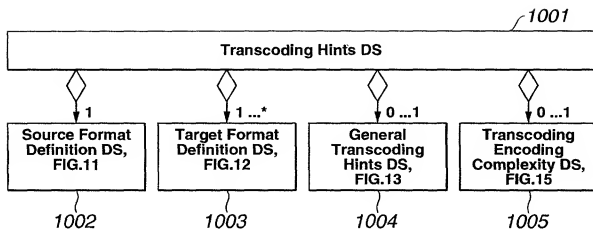


FIG.10

- 1: bitrate <int>
- 2: size_of_pictures <2*int>
- 3: number_of_frames_per_second <int>
- 4: pel_aspect_ratio <float>
- 5: pel_colour_depth <int>
- 6: usage_of_progressive_interlaced_format <1 bit>
- 7: usage_of_frame_field_pictures <1bit>
- 8: compression method <int>
- 9: one out of list {MPEG-1, MPEG-2, MPEG-4, DV, H.263, H.261, }
- 10: { further parameters for compression method }
- 11: GOP_structure (Runlength coding)

FIG.11

11/13

```

1: bitrate <int>
2: size_of_pictures <2*int>
3: number_of_frames_per_second <int>
4: pel_aspect_ratio <float>
5: pel_colour_depth <int>
6: usage_of_progressive_interlaced_format <1 bit>
7: usage_of_frame_field_pictures <1bit>
8: compression_method <int>
9:     one_out_of_list {MPEG-1, MPEG-2, MPEG-4, DV, H.263, H.261, .... }
10:    { further parameters for compression method }
11:    GOP_structure (Runlength coding)

```

FIG.12

```

1: use_region_of_interest_DS: <1bit>
2:     region_of_interest_DS:
3:         shape_D: select one or {boundary_box_D, MB_shape_D, shape_D}
4:         motion_trajectory_D
5:
6: use_editing_effects_transcoding_hints_DS: <1bit>
7:     camera_flash {frame1, frame2, .... framek} <k*int>
8:     cross_fading {(start_frame, end_frame), ...} <k*(<int>, <int>)>
9:     black_pictures {(start_frame, end_frame), ...} <k*(<int>, <int>)>
10:    fade_in {(start_frame, end_frame), ...} <k*(<int>, <int>)>
11:    fade_out {(start_frame, end_frame), ...} <k*(<int>, <int>)>
12:    abrupt_change {frame1, frame2, .... framek} <k*int>
13:
14: use_motion_transcoding_hints_DS: <1 bit>
15:     number_of_regions: <int>
16:     for_every_region:
17:         is_region_rectangular_shaped (y/n): <1bit>
18:         if_arbitrarily_shaped: use region D for this region
19:         describe_parametric_object_motion_for_this_region

```

FIG.13

12/13

1: start_frame <int>
2: nframes <int>
3: I_frame_location:
4: select_one_out_of_the_following: <2 bit>
5: first frame (default)
6: list of frames {frame1, frame2, , framek} <k*int>
7: first_frame_and_every_k_frames <int>
8: no_I_frame
9: quantizer_scale <int>
10: target_bitrate <int>
11: target_min_bitrate <int>
12: target_max_bitrate <int>
13: use_transcoding_states (y/n) <1 bit>
14: transcoding_state_nr <int>
15: add_new_transcoding_state (y/n) <1 bit>
16: if yes: {list of parameters}
17: remove_transcoding_state (y/n) <1 bit>
18: if yes: state_nr <int>
19: use_encoding_complexity_description (y/n) <1 bit>
20: if yes: encoding_complexity_description_scheme

FIG.14

13/13

```

1: use_feature_points (y/n) <1bit>
2:   select_feature_point_method <2 bits>
3:     number_of_new_feature_points <nframes * int>
4:     feature_point_metrics {mean, max, min, var, stddev} <5* int>
5: use_equation_description (y/n) <1bit>
6: use_motion_description (y/n) <1bit>
7:   select_motion_method <4 bits>
8:     param_k_motion <nframes * k * int>
9:     motion_metrics {min, max, sum, var, stddev} <5*int>
10:    block_motion_field < nframes*int*size_x*size_y / (m*m) >
11: use_texture_edge_metrics (y/n) <1bit>
12:   select_texture_edge_metrics <4 bits>
13:     DCT_block_energy <size_y*size_x*nframes*int/64>
14:     DCT_block_activity <size_y*size_x*nframes*int/64>
15:     DCT_energy_metric {mean, min, max, sum, var, stddev} <6*int>
16:     DCT_activity_metric {mean, min, max, sum, var, stddev} <6*int>

```

FIG.15

```

1: M: I/P distance <int>
2: bitrate_fraction_for_I <float>
3: bitrate_fraction_for_P <float> /* bitrate_fraction of B is rest to 100 %
4: quantizer_scale_ratio_I_P <float>
5: quantizer_scale_ratio_I_B <float>
6: if_frame: /* see target format transcoding hints */
7:   X_I, X_P, X_B <3*int> /* frame_vbv_complexities */
8: if_field:
9:   X_I_top, X_P_top, X_B_top <3*int> /* field_top_vbv_complexities */
10:  X_I_bot, X_P_bot, X_B_bot <3*int> /* field_bottom_vbv_complexities */

```

FIG.16